Remarks

The Office Action dated November 6, 2003 has been noted, and its contents carefully studied. In light of the following arguments, reconsideration of the rejection under 35 U.S.C. 103 is courteously requested.

Reconsideration of the finality of the rejection is requested. The Examiner has asserted that Applicants' amendments necessitated the new grounds of rejection. It is respectfully requested that this is not the case. More specifically, the amendments previously made merely further clarify or define features of the invention already initially in the claims. Thus, the new ground of rejection could have been made in the prior Office Action. As such, it is respectfully urged that the finality of the rejection be withdrawn.

To further facilitate the Examiner's reconsideration, a summary of the invention is again presented herein, as recited in the claims which were previously amended in response to the prior Office Action.

In one aspect, the invention is a method of searching file access system for a requested file. Claim 1 clearly recites that part of the method involves establishing a field in a directory i-node memory structure for files corresponding to a directory cache hash table with a field contacting a pointer to the directory cache hash table. The cache hash table is characterized as having an array of hash buckets which point to a list of files which may correspond to a specific i-node. The directory cache hash table is searched for a requested file by hashing the file i-node to a specific bucket which contains a list of files that may correspond to the requested file i-node, and if the bucket contains a matching file name, pointing to where the name of requested file is stored.

In a further aspect, as decided in Claim 4, the directory is read into buffer cache with the directory having a storage device representation. The directory is converted from the storage device representation to a faster representation, with the faster representation representing a

layout of the directory with an array of hash buckets which point to a list of files which may correspond to a specific i-node.

In a still further aspect, the storage device representation is maintained for backwards compatibility with pre-existing file access systems.

In an alternative aspect, the invention recites a computer server system, including at least one processor allocating memory for buffer cache and directory cache. The processor converts directly from a storage device layout to a faster representation which includes an array of hash buckets which point to a list of files which may correspond to a specific i-node.

Yet still further, the invention is also directed to a network storage system which includes at least one processor allocating memory for buffer cache and directory cache, with the processor converting directories from a storage layout to a faster representation which includes an array of hash buckets which point to a list of files which may correspond to a specific i-node. The faster representation includes a pointer from a directory i-node memory structure to an associated hash table.

Having thus generally discussed the invention, notwithstanding the Examiner issuing a new ground of rejection, it will become readily apparent to the Examiner that the invention is not rendered obvious from the cited references, based on the following discussion of the references presented herein for the Examiner's kind consideration.

U.S. Patent 5,151,989 to Johnson et al.

U.S. Patent 5,151,989 to Johnson et al. (Johnson) discloses directory cache management in a distributed data processing system. When a lookup request occurs, the file is identified by a file handle. The file handle contains the device number, the i-node number and i-node generation fields (column 23, lines 52-59). The kind of data stored in directory cache is information about nodes. For each node there can be multiple entries with the same name. This is possible because the same name may have occurred in two different directories in a node. Three elements, node ID, directory file handle, and name, together form the search key that is

used when searching the directory cache. In order to speed up searching of the directory cache, the actual data structure used is more complex (column 17, lines 46-60). Functions are defined that hash any node ID, directory file handle, and path name part into an index into the hash table. All entries that hash into the same location into the hash table are chained together in the same previous-next list that is pointed to by the pointer in the hash table at that location (column 18, lines 17-22).

The above teachings of the Johnson patent are direct quotes from the sections cited by the Examiner and not representative of what is claimed, for example, in Claim 1. Accordingly, it is respectfully urged that the propositions for which the Examiner has cited Johnson for are incorrect. There is nothing in Johnson which teaches establishing a field in a <u>directory i-node memory structure</u> for files corresponding to a directory cache hash table, with the field containing a pointer to the directory cache hash table. There is no teaching in Johnson or suggestion therein of a directory i-node memory structure utilized in accordance with Applicants' invention. Yet still further, there is no teaching or suggestion of searching the directory cache hash table for a requested file by hashing the file i-node <u>to a specific bucket</u> which contains a list of files that may correspond to the requested file i-node, followed by the additional step of, if the bucket contains a matching file name (consider that the bucket may not contain the matching file name), then pointing to where the name of the requested file is stored.

As to Claim 4, although the Examiner has asserted that Johnson teaches reading a directory into cache, in accordance with the invention the directory having a storage device representation is read on the disk, and then the directory is converted from the storage device representation to a faster representation. This is simply not the case in Johnson.

More specifically, the cited sections of the patent teach instead: read the file identified by the i-node number associated with dir1 (this is the next directory in the path); when a file is opened, its i-node is read from the disk and a subset of the i-node information, together with some additional information, is stored in the i-node table; in this example, notice that information about three different nodes is being cached; nodes with IDs 12, 20, and 52. For node 12, there

are two entries with the same name, "bin". This is possible because this name "bin" occurred in two different directories in node 12. There may be directories that have the handles of 177 and 123, in this case, perhaps representing /bin and /usr/bin, two common directories in AIXtm systems.

The three elements, node ID, directory file handle, and name, together form the search key that is used when searching the directory cache. In order to speed up searching of the directory cache, the actual data structure used is more complex. Any of the commonly used methods for structuring information for file searching may be used. The method illustrated in Figs. 19 and 20 uses a hash table to speed up the searching of the directory cache.

In this regard, the above-described teaching has nothing to do with Applicants' claims requiring converting the storage device representation to a faster representation and at the same time maintaining the storage device representation for backwards compatibility. In fact, there is no conversion indicated in the teachings of Johnson.

Similarly, Claim 7 of the patent which has been cited for the proposition of teaching the representing of a layout of the directory with an array of hash buckets simply fails to support this interpretation. Instead, the claim, is a method of saving at a first system a first unit of directory information about a second system in response to an inquiry by the first system regarding a directory structure of the second system. Thereafter, the claim calls for automatically sensing an additional unit of directory information from the second system to the first system in response to subsequent change in the directory structure of the second system, and detecting an inhibiting condition associated with the network of data processing systems. This nothing to do with the claimed invention.

It is respectfully urged that what the Examiner has done in citing Johnson to support the 35 U.S.C 103 rejection is simply a hindsight interpretation of the teachings of Johnson in an attempt to arrive at the claimed invention. The stubborn fact remains that there is nothing in Johnson alone, or in combination with the other references, that anticipates or suggests the

invention as discussed herein both with reference to the independent claims, and as further defined by the claims which are dependent from those independent claims.

The other references cited to reject the claims add nothing further to the teachings of Johnson and in fact have already been discussed. Nonetheless, to complete the response herein, these references will again be discussed herein.

U.S. Patent No. 5,666,532 to Saks et al.

U.S. Patent No. 5,666,532 to Saks et al. (Saks) has been previously discussed in a prior Response. The Examiner has asserted that Saks teaches a computer method for order operations in which allocating memory for buffer cache is taught.

In fact what Saks teaches is a computer system having data organized in files with secondary storage for storing files, having one or more types of file subsystems for controlling transfer of files between primary and secondary storage (abstract). This has nothing to do with Applicants' claimed invention. In fact, Saks merely teaches a conventional block device model which is usually applied to disks and tapes. The purpose of the block model is to allow the use of buffering in order to obtain reduced i/o traffic. The sizes of buffers used in performing transfers from or to block devices are some convenient multiples of 512, which is the size of typical disk block. The buffering for block i/o transfers is performed by using some or all pages from the page cache 41, or by a dedicated pool of memory called buffer cache 40, and frequently by employing both the page cache and a buffer cache.

Use of buffer and cache in Saks adds nothing to the teachings of Johnson and is merely a conventional prior art technique which is used in systems of that type. Again it is respectfully urged that the combination of Johnson and Saks fail to anticipate or render obvious Applicants' claimed invention, contrary to the Examiner's stated assertions.

U.S. Patent No. 5,778,430 to Ish et al.

U.S. Patent No. 5,778,430 to Ish et al. (Ish) teaches a computer disk cache management method and apparatus which employs a least-recently-used with aging method to determine a best candidate for replacement as a result of the cache miss. If a directory check indicates that a block is not within the cache, then the block is performed and the block is loaded from the direct access storage device into the cache after a space of suitable size is made within the cache to hold the block.

This not the same as teaching hashing directories according to a size of the directory. Instead, what Ish teaches is if a block is not within a cache, then allocating sufficient space within the cache to hold the block.

Again it is respectfully urged that the Office Action has not correctly interpreted the teachings of Ish and for the reasons previously set forth and as further discussed herein, it is urged Ish fails to add anything to the teachings of Saks and Johnson to anticipate or render obvious Applicants' claimed invention.

As already previously pointed out, Ish teaches nothing more that conventional caching of data with a specific mechanism for replacing data which is least frequently accessed, by new data which may be accessed more commonly. The stubborn fact remains, that it is only after knowledge of Applicants' claimed invention in a hindsight interpretation of the references that the Examiner has been able to arrive at the rejection. This is clearly impermissible under the law and a well stated principle in numerous cases from the Federal Circuit throughout the years.
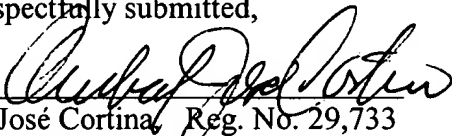
Accordingly, it is respectfully urged that the claims are clearly allowable over the cited art and that the application should be passed to issuance.

Nonetheless, should the Examiner have any comments, questions or suggestions of a nature necessary to expedite prosecution of the application or to resolve any outstanding issues, he is courteously requested to telephone the undersigned at the number listed below.

Dated: February 5, 2004

Respectfully submitted,

A. José Cortina, Reg. No. 29,733
Daniels Daniels & Verdonik, P.A.
P.O. Drawer 12218
Research Triangle Park, NC 27709
Voice 919.544.5444
Fax 919.544.5920
Email jcortina@d2vlaw.com

Enclosures

F:\CL\1301-003\Prosecution\RespFinalOA.doc